

AMENDMENT TO THE CLAIMS

1-36 (canceled)

1 37. (previously presented): A method for providing a capability to securely
2 update information stored in a plurality of computer systems, wherein the method
3 comprises:

4 forming a protected partition within a hard drive of each of the computer
5 systems

6 storing, within nonvolatile storage of each computer system in the plurality
7 of computer systems, a setup password, an operating system, and an
8 initialization routine to execute within a processor of the computer system after
9 power on of the computer system, wherein the initialization routine includes
10 instructions causing the protected partition to be locked before the operating
11 system is loaded, and wherein instructions causing information stored within the
12 a predetermined location to be written within the protected partition after
13 predetermined security procedures using the setup password have occurred but
14 before the protected partition is locked;

15 establishing a network connecting each computer system in the plurality of
16 computer systems with a server system;

17 generating an update partition file within the server system;

18 transmitting the update partition file over the network to each computer
19 system in the plurality of computer systems; and

20 storing the update partition file within the predetermined location of each
21 computer system in the plurality of computer systems.

1 38. (currently amended): The method of claim 37, wherein the initialization
2 routine includes instructions causing the processor of the computer system to
3 perform a method including:

4 comparing information stored in the protected partition with information

5 from the update partition file stored within the predetermined location;
6 when a portion of the information stored in the protected partition is found
7 partition is found to match a portion of the information stored within the update
8 partition file, overwriting the portion of the information stored in the protected
9 partition with the portion of the information stored in the update partition file if
10 space around the portion of the information stored in the protected partition is
11 sufficient;
12 when a portion of the information stored in the protected partition is not
13 found to match a portion of the information stored within the update partition file,
14 writing the portion of the information stored within the update partition file to
15 append to the information stored in the protected partition if space within the
16 protected partition is sufficient; and
17 locking the protected partition to prevent further modification of
18 information stored within the protected partition.

1 39. (previously presented): The method of claim 38, wherein
2 a flag bit is set in non-volatile storage within the computing system when
3 the update partition file is stored at a predetermined location in non-volatile
4 storage within the computing system, and
5 determining whether the update partition file is stored within the
6 computing system for updating the protected partition is performed by
7 determining whether the flag bit is set.

1 40. (previously presented): The method of claim 38, wherein
2 the method additionally comprises, after determining that the update
3 partition file is stored within the computing system for updating the protected
4 partition, verifying whether the update partition file has been generated by the
5 server system, and

6 the portion of the update partition file is written to the protected partition
7 only following verification that the update partition file has been generated by
8 the server system.

1 41. (currently amended): The method of claim 40, wherein verification that the
2 update partition file has been generated by the server system includes:

3 forming a first message digest by applying a hash algorithm to a portion
4 of the update partition file;

5 forming a second message digest by decrypting a digital signature within
6 the update partition file using a public key of the server system; and[[:]]

7 determining that the first and second message digests are identical.

1 42. (previously presented): The method of claim 40, wherein

2 the predetermined setup procedures include verifying that the update
3 partition file has been generated by the server system includes signing an
4 encrypted portion of the update partition file with a public key of the trusted
5 server system, and

6 the encrypted portion of the update partition file has been prepared by
7 signing, with a private key of the server system, a result of the application of an
8 algorithm to data including a version of the setup password accessed by the
9 server system.

1 43. (previously presented): The method of claim 42, wherein

2 the data includes the version of the setup password appended to a
3 portion of the update partition file ,

4 said algorithm is a hash algorithm generating a message digest, and

5 verifying that the update partition file has been generated by the server
6 system includes applying the hash algorithm to the setup
7 password stored within the computing system appended to a portion of the
8 update partition file to generate a first version of a message digest and

9 comparing the first version of the message digest with a second version of the
10 message digest obtained by signing the encrypted portion of the update
11 partition file.

1 44. (previously presented): The method of claim 38, wherein
2 the update partition file includes a plurality of entries and a plurality of
3 encrypted elements,
4 each entry within the plurality of entries includes information to be stored
5 at a different location within the protected partition,
6 each encrypted element within the plurality of encrypted elements is
7 associated with an entry in the plurality of entries,
8 the method additionally comprises, following determining that the update
9 partition file is stored within the computing system for updating the protected
10 partition, verifying whether each entry in the plurality of entries within the update
11 partition file has been generated by the server system, and
12 each entry in the plurality of entries within the update partition file is
13 written to the protected partition only following verification that the entry has
14 been generated by the server system.

1 45. (currently amended): The method of claim 44, wherein verifying that ~~the~~
2 each entry in the plurality of entries within the update partition file has been
3 generated by the server system includes:
4 forming a first message digest by applying a hash algorithm to the entry;
5 forming a second message digest by signing the encrypted element
6 associated with the entry using a public key of the server system; and;
7 determining that the first and second message digests are identical.

1 46. (currently amended): The method of claim 44, wherein verifying that ~~the~~
2 each entry in the plurality of entries within the update partition file has been
3 generated by the server system includes signing the encrypted element

4 associated with the entry with a public key of the server system, and the
5 encrypted element of the update partition file has been prepared by signing,
6 with the private key of the server system, a result of the application of an
7 algorithm to data including a version of the setup password accessed by the
8 server system.

1 47. (currently amended): The method of claim 46, wherein
2 the data includes the version of the setup password appended to ~~a~~the
3 entry,
4 the algorithm is a hash algorithm generating a message digest, and
5 verifying that the entry has been generated by the server system includes
6 applying the hash algorithm to the setup password stored within the computing
7 system appended the entry to generate a first version of a message digest and
8 comparing the first version of the message digest with
9 a second version of the message digest obtained by signing the encrypted
10 element.

1 48. (previously presented): The method of claim 44, wherein
2 information stored in the protected partition is compared to each entry in
3 the plurality of entries within the update partition file,
4 when a portion of the information stored in the protected partition is found
5 to match the entry, the portion of the information stored in the protected partition
6 is overwritten with the entry if space around the portion of the information stored
7 in the protected partition is sufficient, and
8 when a portion of the information stored in the protected partition is not
9 found to match the entry, the entry is appended to the information stored in the
10 protected partition if space within the protected partition is sufficient.

1 49. (currently amended): The method of claim 38 ~~48~~48, wherein
2 the method additionally comprises receiving an input signal from a

3 keyboard of the computing system and comparing the input signal with a signal
4 corresponding to a setup password stored in non-volatile storage within the
5 computing system, and

6 the protected partition is left unlocked if the input signal matches the
7 signal corresponding to the setup password.

1 50. (previously presented): An interconnected system for providing updated
2 information in a secure manner, wherein the interconnected system comprises:

3 a network;

4 a server system connected to the network and programmed to generate
5 an update partition file and to transmit the update partition file over the network;
6 and

7 a computer system connected to the network, wherein the computer
8 system includes a processor, non-volatile data storage including a hard drive
9 having a protected partition, wherein the processor is programmed to receive the
10 update partition file from the network and to store the update partition file in a
11 predetermined location within the nonvolatile data storage outside the protected
12 partition, and wherein the nonvolatile data storage stores an operating system
13 and an initialization routine, executing within the processor after power on of the
14 computer system, including instructions causing the protected partition to be
15 locked before the operating system is loaded, and instructions causing
16 information stored within the predetermined location to be written within the
17 protected partition after predetermined security procedures have occurred but
18 before the protected partition is locked.

1 51. (currently amended): The interconnected system of claim 50, wherein the
2 initialization routine includes instructions causing the processor of the computer
3 system to perform a method including:

4 comparing information stored in the protected partition with information
5 from the update partition file stored within the predetermined location;

6 when a portion of the information stored in the protected
7 partition is found to match a portion of the information stored within the update
8 partition file, overwriting the portion of the information stored in the protected
9 partition with the portion of the information stored in the protected partition if
10 space around the portion of the information stored in the protected partition is
11 sufficient;

12 when a portion of the information stored in the protected partition is not
13 found to match a portion of the information stored within the update partition file,
14 writing the portion of the information stored within the update partition file to
15 append to the information stored in the protected partition if space within the
16 protected partition is sufficient; and

17 locking the protected partition to prevent further modification of
18 information stored within the protected partition.

1 52. (previously presented): The interconnected system of claim 51, wherein
2 a flag bit is set in non-volatile storage within the computing system when
3 the update partition file is stored at a predetermined location in non-volatile
4 storage within the computing system, and
5 determining whether the update partition file is stored within the
6 computing system for updating the protected partition is performed by
7 determining whether the flag bit is set.

1 53. (previously presented): The interconnected system of claim 51, wherein
2 the method additionally comprises, after determining that the update
3 partition file is stored within the computing system for updating the protected
4 partition, verifying whether the update partition file has been generated by a
5 trusted server system, and
6 the portion of the update partition file is written to the protected partition
7 only following verification that the update partition file has been generated by
8 the server system.

1 54. (previously presented): The interconnected system of claim 53, wherein
2 verification that the update partition file has been generated by the server
3 system includes:

4 forming a first message digest by applying a hash algorithm to a portion
5 of the update partition file;

6 forming a second message digest by decrypting a digital signature within
7 the update partition file using a public key of the server system; and;

8 determining that the first and second message digests are identical.

1 55. (previously presented): The interconnected system of claim 53, wherein
2 the predetermined setup procedures include verifying that the update
3 partition file has been generated by the server system includes signing an
4 encrypted portion of the update partition file with a public key of the trusted
5 server system, and

6 the encrypted portion of the update partition file has been prepared by
7 signing, with a private key of the server system, a result of the application of an
8 algorithm to data including a version of a setup password accessed by the
9 server system.

1 56. (previously presented): The interconnected system of claim 55, wherein
2 the data includes the version of the setup password appended to a
3 portion of the update partition file,

4 the algorithm is a hash algorithm generating a message digest, and

5 verifying that the update partition file has been generated by the server
6 system includes applying the hash algorithm to the setup
7 password stored within the computing system appended to a portion of the
8 update partition file to generate a first version of a message digest and
9 comparing the first version of the message digest with a second version of the

10 message digest obtained by signing the encrypted portion of the update
11 partition file.

1 57. (previously presented): The interconnected system of claim 51, wherein
2 the update partition file includes a plurality of entries and a plurality of
3 encrypted elements,
4 each entry within the plurality of entries includes information to be stored
5 at a different location within the protected partition,
6 each encrypted element within the plurality of encrypted elements is
7 associated with an entry in the plurality of entries.
8 the method additionally comprises, following determining that the update
9 partition file is stored within the computing system for updating the protected
10 partition, verifying whether each entry in the plurality of entries within the update
11 partition file has been generated by the server system, and
12 each entry in the plurality of entries within the update partition file is
13 written to the protected partition only following verification that the entry has
14 been generated by the server system.

1 58. (previously presented): The interconnected system of claim 57, wherein
2 verifying that the each entry in the plurality of entries within the update partition
3 file has been generated by the server system includes:
4 forming a first message digest by applying a hash algorithm to the entry;
5 forming a second message digest by signing the encrypted element
6 associated with the entry using a public key of the server system; and;
7 determining that the first and second message digests are identical.

1 59. (previously presented): The interconnected system of claim 57, wherein
2 verifying that the each entry in the plurality of entries within the update partition
3 file has been generated by the server system includes signing the encrypted
4 element associated with the entry with a public key of the server system, and

5 the encrypted element of the update partition file has been prepared by signing,
6 with the private key of the server system, a result of the application of an
7 algorithm to data including a version of a setup password accessed by the
8 server system.

1 60. (currently amended): The interconnected system of claim 59, wherein
2 the data includes the version of the setup password appended to a the
3 entry,
4 said algorithm is a hash algorithm generating a message digest, and
5 verifying that the entry has been generated by the server system includes
6 applying the hash algorithm to the setup password stored within the computing
7 system appended the entry to generate a first version of a message digest and
8 comparing the first version of the message digest with a second version of the
9 message digest obtained by signing the encrypted element.

1 61. (previously presented): The interconnected system of claim 57, wherein
2 information stored in the protected partition is compared to each entry in
3 the plurality of entries within the update partition file,
4 when a portion of the information stored in the protected partition is found
5 to match the entry, the portion of the information stored in the protected partition
6 is overwritten with the entry if space around the portion of the information stored
7 in the protected partition is sufficient, and
8 when a portion of the information stored in the protected partition is not
9 found to match the entry, the entry is appended to the information stored in the
10 protected partition if space within the protected partition is sufficient.

1 62. (currently amended): The interconnected system of claim 54 61, wherein
2 the method additionally comprises receiving an input signal from a

3 keyboard of the computing system and comparing the input signal with a signal
4 corresponding to a setup password stored in non-volatile storage within the
5 computing system, and
6 the protected partition is left unlocked if the input signal matches the
7 signal corresponding to the setup password.